Scientific Research – How Bazaar!

James Myers
Pacific Northwest National Laboratory
Jim.Myers@pnl.gov

Science is often presented as a majestic pursuit, researchers standing "on the shoulders of giants" (Isaac Newton) to collaboratively build an ever more complete understanding of 'everything'. And we, as computing researchers and developers, have bought it. We build common infrastructures, shared middleware, and integrated collaboration and problem solving environments that provide the 'perfect' scaffolding for the orderly extension of science's towers of theory, yet our successes (though substantial) don't approach the potential we perceive. Scientists are slow to adopt our approaches and products, few pilot projects survive into production, and domain scientists complain that we "just don't get it". (The feeling can be mutual of course.) What aren't people getting? And what can we, as computing researchers, do about it?

Simply put, science is not conducted as it is described in text books. At the most fundamental level, the popular view of science as a steady community-driven march towards better understanding is a fallacy. Thomas Kuhn argued in "The Structure of Scientific Revolutions" that, not only was science not a direct march toward the truth, it was also driven by culture as much as, or more than, by logic; theories aren't always replaced by 'better' ones and 'incorrect' theories can persist long after evidence is available to refute them. While one can argue that science practice is simply irrational and researchers make choices, including choices about information technologies, completely unpredictably, this really goes too far. Clearly, one must accept that science-in-action is much messier than science-as-history. Beyond that, if the cultural aspects of science are not unpredictable, but are actually driven by a 'cultural economics' that can be understood, one can design software that accounts for scientific culture.

What is the right way to model science's cultural economics? As a bazaar! The bazaar metaphor, popularized in computing to describe the open source model, also applies well to science. (Interestingly, one can make further connections between modern programming practice and scientific research and consider extreme programming (XP) as an application of the scientific method – see http://www.agiledevelopmentconference.com/2003/files/P6Paper.pdf.) Research involves a lively competition of ideas that drives progress, but the 'business models' chosen by different groups depends on the specifics of the market in their sub-domain and their personalities. Constraints such as the ratio of fixed to marginal costs, the market size, and the ability to leverage a business advantage across multiple markets, vary across sub-disciplines and with time. Accepting this, it isn't hard to see why common software infrastructures often fail to spread across the entire scientific bazaar. It isn't from ignorance or some act of stubbornness on the part of researchers – something that can at best be solved by sociologists and through education and training. Rather, it is a natural consequence reflecting the variation in value of the infrastructure across sub-domains, lifecycle versus acquisition costs (what is the opportunity cost of a graduate student learning standard data formats?), and a search for 80% solutions that have 20% of the cost. While one can argue that there are tragedies of the commons involved, it is also important to ask how the realities of science can be addressed through new approaches and designs.

One need only look at the evolution of shared science infrastructure projects such as collaboratories, grids, problem solving environments, and community databases over the last decade to realize that changes are already being made to address scientific reality: shifts from

monolithic to component-based to service-based architectures, increasing use of open APIs and protocols, increasing recognition of the value of focused pilot projects, etc. However, their adoption is perhaps more akin to the addition of epicycles to circular orbits than an adoption of elliptical model. Where might we look for our 'paradigm shift' in supporting the science bazaar? As argued elsewhere at this meeting (Larry Rahn et. al., "Can collaboratories relieve the predicament of the modern scientists?"), we need to consider adoptability and adaptability up front – in the same way security is (or should be). As discussed below, there are a number of current and emerging technologies and design patterns to support this. Taken together, I believe these suggest a means for our community to support science as it is practiced, and as researchers would like to practice it.

Toolkit approaches start in this direction, but they don't go far enough. There are numerous examples where, for example, instrument control has been integrated with a collaborative suite. While researchers can see benefits and pilot projects are successful, maintaining the integration in the long term, and simply the requirement for users to start a collaboration tool to run the instrument solo (or to learn two interfaces), limit adoption and long-term impact. Consider instead, designing the instrument to support a general publish/subscribe mechanism, and having the collaborative suite discover its existence and include it. The instrument can now be run stand-alone, and the event mechanism can be used to support collaboration, records keeping, and/or integration into a PSE, with its adoption and upkeep supported by multiple drivers.

Similarly, consider common data formats. While they allow collaboration, they place the costs of compliance on the producer, while benefits accrue to the consumer of the data. Consider instead a mechanism such as the Data Format Description Language, which allows arbitrary formats to be described after the fact, coupled with data virtualization services, which can use DFDL descriptions to pull data into any desired format on demand (both being standardized now in the Global Grid Forum). Researchers can choose optimum formats for their work, delay the costs of coordination until its value can be realized, shift these costs to other parties, and easily participate in multiple communities with conflicting standards.

Such thinking can also be applied to services and workflows. Instead of requiring adoption of a standard design to handle the most demanding case, one can use discovery and translation mechanisms to achieve global capabilities while allowing smaller groups to drive towards lighter-weight mechanisms optimized for their needs. In this view, metadata shifts from being a way to describe data (a fairly un-contentious definition until one tries to distinguish data and metadata) to a light-weight, aspect-oriented mechanism for managing and translating between the data models of multiple related processes.

In some sense, these techniques are just logical extensions of what is done today. In another, they represent a new level of computing capability – the start of the "semantic grid" perhaps. However, from a philosophical perspective, they might just be more. They might be the first time that a proposed cyberinfrastructure can be made that is "as simple as possible -- but no simpler" (Albert Einstein) than what is needed to enable the scientific bazaar to tackle the challenges of next-generation, systems-oriented research. If so, history may describe our epicycles of progress much differently than we would.